Messier: A High-Resolution Corpus for Cross-Benchmark Agent Evaluation

Anonymous ACL submission

Abstract

Evaluating AI agents in interactive environments is hindered by fragmented tasks, scaffolds, verifiers, and scoring rules. Existing efforts focus on narrow settings, remain limited in scale, or require costly reruns, leaving much of the empirical record incomparable. We introduce MESSIER, a unified corpus of 957,253 records that span 30 benchmarks, 714 agents, 11,891 tasks, and 74,205 verifiers. MESSIER consolidates public benchmark scores and supplements them with five-agent runs across six underrepresented professional and scientific domains, including a recent legal benchmark. Each record is standardized by model, scaffold, environment, task, verifier, and aggregation rule, with SOC/NAICS classifications for occupational and industry analysis. Using this corpus, we show frontier progress is uneven across benchmark types, with "function calling" saturated, "programming" improving the fastest, and "enterprise workflows" remaining the most challenging. Furthermore, counterfactual rescoring shows that strict all-pass aggregation in multi-verifier tasks can obscure progress and artificially alter agent rankings. From these standardized records, we derive capability scales that align with Epoch's Evaluation Capability Index rankings at Spearman $\rho = 0.81$ and can be specialized by domain, occupation, action space, or verifier type. MESSIER provides a foundational, reusable infrastructure for agent capability scaling, benchmark auditing, and fine-grained analysis of evaluation failures.

1 Introduction

Agent benchmarks are now a central instrument for measuring how language models act in interactive environments. They test whether these systems can complete delegated professional work under domain constraints, using evidence and tools to produce outputs that can affect scientific claims (Bragg et al., 2025), clinical records (Jiang et al., 2025), legal matters (Grupen et al., 2026), software systems (Jimenez et al., 2024), or financial decisions (Bigeard et al., 2025; Zhang et al., 2026).

This proliferation has made results increasingly difficult to compare across the field. Benchmarks vary in domain, agent definition, environment, action space, and success condition (Kapoor et al., 2025). Final scores further hide these differences by compressing complex behaviors into a single number. A math task might be evaluated with one binary check, whereas a multi-step business workflow may use a suite of tests aggregated into one score. Although both yield a score between 0 and 1, a pass in one setting reflects a different kind of success than a pass in the other (Burnell et al., 2023; Yang and Wang, 2026; Sha and Zhao, 2026).

To make heterogeneous results more comparable, recent frameworks pursue unified measurement. Epoch's Evaluation Capability Index (ECI) (Ho et al., 2025) applies Item Response Theory (IRT) (Rasch, 1960) to map diverse benchmark scores onto a shared scale of model capability and task difficulty. A complementary line of work, METR (Kwa et al., 2026), measures agent capacity by task horizon length, showing that model execution limits have doubled roughly every seven months since 2019.

The data underlying these scales remain incomplete. Epoch releases benchmark-level inputs rather than individual task outcomes and focuses primarily on coding and math, while METR provides per-task granularity for only 170 tasks (Kwa et al., 2026). Consequently, extending these analyses often requires researchers to construct evaluation corpora from scratch before rerunning costly agent evaluations, with recent large-scale sweeps costing roughly $40,000 (Kapoor et al., 2025). Existing meta-collections partially address this gap, but either focus on narrow agentic settings such as tool use, center on static non-agentic benchmarks, or remain limited in scale (Liang et al., 2022; Sri-

Resource	Agentic	Benchmarks	Environments	Models	Scaffolds	Tasks
HELM (Liang et al., 2022)	✗	42	N/A	30	N/A	30,000
METR (Kwa et al., 2026)	✓	3	3	13	N/A	170
Epoch ECI (Ho et al., 2025)	✓	37	N/A	200	N/A	N/A
TOUCAN (Xu et al., 2025)	✓	N/A	495	3	2	1.5M
HAL (Kapoor et al., 2025)	✓	9	N/A	9	∼10	∼1,500
BRIDGE (Liu et al., 2026)	✓	4	N/A	176	∼70	752
Agent Psych. (Ge et al., 2026)	✓	4	N/A	275	∼90	1,421
MESSIER	✓	30	5,360	345	205	11,891

Table 1: **Comparison with related collections of LLM evaluation data.** Counts are as reported by each resource at release; ∼ denotes approximate values inferred from paper. N/A means the information is not reported or applicable.

vastava et al., 2023; Lù et al., 2025; Xu et al., 2025). At the same time, emerging concerns such as reward hacking (Thaman, 2026; Atinafu and Cohen, 2026; Gabor et al., 2025; Tan et al., 2025), evaluation awareness (Needham et al., 2025; Hua et al., 2025), and sandbagging (Van Der Weij et al., 2025; Gasteiger et al., 2025) further highlight the need for open, inspectable data.

We introduce MESSIER, a corpus of 960 k trial outcomes consolidating 24 public releases and new five-agent runs on six benchmarks spanning quantum algorithm design, legal review, clinical workflows, and other under-evaluated professional domains. With SOC/NAICS task classifications and partial trace coverage, MESSIER supports analyses of agent behavior, verifier design, and evaluation failure modes. Our contributions are:

- **A unified agent evaluation corpus** consolidating 30 benchmarks (with new runs on six), 714 agents, 11,891 tasks, and 74,205 verifiers (§4), providing both macro- and micro-scale views of agent performance.
- **Evidence that aggregation rules change measured capability**, enabled by verifier-level records that support counterfactual rescoring. For example, all-pass scoring can report near-zero task success even when agents satisfy many criteria on average.
- **Capability indices from open data** that correspond to Epoch ECI rankings at Spearman $\rho = 0.81$ (§5.3) and support slicing by occupation, action space, or verifier design.

By enabling researchers to run, share, and contribute evaluations within a single dataset, MESSIER stands to reduce redundant compute.[1]

[1]Code and sample data: anon.4open.science/messier-d3

2 Related Work

Agent benchmarks. Evaluating models within interactive environments shifts the testing paradigm away from static text pairs toward sequential execution, where an agent observes states over multiple turns, selects actions, modifies files, or invokes tools before a verifier outputs an outcome. Recent benchmarks instantiate this setup across several environment families, including software engineering (Jimenez et al., 2024; Shetty et al., 2026; Jain et al., 2025), web navigation (Xue et al., 2025; Wei et al., 2025; He et al., 2024), terminal and operating-system tasks (Merrill et al., 2026; Xie et al., 2024), tool-use and function-calling environments (Patil et al., 2025; Barres et al., 2025; Wang et al., 2025b), security tasks (Zhang et al., 2025), economic work (Patwardhan et al., 2025; Xu et al., 2026), and professional domains such as science, medicine, and law (Grupen et al., 2026; Bragg et al., 2025; Jiang et al., 2025). However, each benchmark dictates its own naming conventions, scaffolding interfaces, and underlying verifier architectures. Consequently, cross-benchmark comparison requires significant manual effort and currently relies on tedious post-hoc consolidation or expensive rerunning.

Evaluation collections. Parallel efforts increasingly focus on cross-benchmark consolidation by assembling heterogeneous execution traces into unified formats. These efforts range from non-agentic text collections such as HELM (Liang et al., 2022) and BIG-bench (Srivastava et al., 2023) to collections built around interactive environments (Liu et al., 2024; Mialon et al., 2024; Ma et al., 2024). However, current repositories remain limited by their specific operational scopes or focus areas. For instance, AgentRewardBench (Lù et al.,

2025) is focused on evaluating trajectory judges, while TOUCAN (Xu et al., 2025) provides synthetic data optimized for agent training rather than empirical capability indexing. The closest precedent to our work is the Holistic Agent Leaderboard (HAL) (Kapoor et al., 2025), which introduces an active, cost-controlled evaluation harness and a corpus of over 21,000 agent rollouts across nine benchmarks. HAL addresses the problem of producing comparable new evaluations under a shared execution protocol. However, much of the empirical record of agent evaluation already exists outside any single harness, dispersed across public releases with incompatible task identifiers, agent metadata, scoring semantics, verifier definitions, and trace formats. As a result, existing results remain difficult to compare, audit, or reuse beyond the settings in which they were originally released.

Heterogeneity in benchmark evaluation. A growing body of literature indicates that aggregate benchmark scores frequently conceal information visible only at the instance level, that is, at individual tasks or verifier outcomes within a benchmark (Schaeffer et al., 2023; Madaan et al., 2024; Lourie et al., 2024). BenchScope (Sha and Zhao, 2026) demonstrates that per-instance analysis reveals roughly five times the latent dimensionality found in macro-level analysis, suggesting that single headline metrics conflate multiple distinct capabilities. Similarly, Yang and Wang (2026) document that language models achieving identical aggregate accuracy can still disagree on 16% to 66% of individual items, creating risks for conclusions drawn purely from macro totals. This risk extends to automated evaluation layers: CARE (Zhao et al., 2026) shows that aggregation under correlated judge errors can distort leaderboard rankings, while SkillsBench (Li et al., 2026a) and ACE-Bench (Chen et al., 2025a) show that final scores are sensitive to benchmark composition, including the mix of domains, skills, and task horizons. If aggregate pass rates collapse heterogeneous task structures and verification rules into a single metric, then macro-level comparisons can misrepresent model capability. We address this limitation with a cross-benchmark corpus that preserves individual task and verifier outcomes.

3 Background

In an interactive evaluation, a large language model (LLM) acts through a scaffold, receives feedback from an environment, and produces a trial scored by one or more verifiers. Across benchmarks, however, the boundaries between task, environment, and verifier are not always drawn consistently. We therefore introduce the framework used in this work and identify where cross-benchmark variation exists.

Agents and environments. An *agent* is a pair $(model, scaffold)$: the model produces text, while the scaffold parses that text into actions, executes them, and returns observations. We treat the scaffold as part of the agent because the same model can yield substantially different benchmark outcomes under different scaffolds (Brand and Denain, 2025; Kapoor et al., 2025).

An *environment* is the system with which the agent interacts. This interaction is often formalized as a partially observable Markov decision process (POMDP) (Kaelbling et al., 1998); for our purposes, the relevant components are the state space $\mathcal{S}$, action space $\mathcal{A}$, and observation space $\mathcal{O}$, which define what the environment can be in, what the agent can do, and what the agent can see. In LLM evaluations, the state is typically an external system, such as a database or codebase, and observations are textual or visual renderings of that state, such as command outputs or web pages.

Tasks and verifiers. A *task* $T = (I, s_0)$ pairs an instruction I with an initial environment state $s_0 \in \mathcal{S}$, and a *trial* τ (also termed a *rollout*) is one execution of an agent on a task. Each task is scored by N *verifiers* $\{v_1, \ldots, v_N\}$, where each $v_i : \tau \to \{0, 1\}$ is typically a programmatic check, an LLM judge, an exact-match check, or a human evaluation. A *trial outcome* is a single verifier output $v_i(\tau)$ (also referred to as a *record*). Benchmarks differ both in which verifier types they use and in how they treat a single scored unit: some group many requirements into one task, while others split them into separate tasks.

When a task has multiple verifiers, the benchmark must decide how their outcomes are collapsed into a single task score. Formally, an aggregation rule $f : \{0, 1\}^N \to \{0, 1\}$ maps verifier outcomes to $S = f(v_1(\tau), \ldots, v_N(\tau))$. In the simplest case, there is only one verifier, so $S = v_1(\tau)$. With *all-pass* grading, every verifier must pass:

$$S = \prod_{i=1}^{N} v_i(\tau).$$

With *threshold* grading, the task passes when at least a benchmark-defined fraction $\theta \in (0, 1]$ of verifiers pass:

$$S = \mathbb{I}\left(\frac{1}{N}\sum_{i=1}^{N} v_i(\tau) \geq \theta\right).$$

The reconciliation problem. Within a single benchmark, these choices are usually straightforward. Across benchmarks, they create a reconciliation problem: scaffolds must be normalized alongside model names, and scores must be interpreted through their verifier and aggregation structure. The map f is many-to-one, so S alone does not recover the verifier outcomes that produced it. This motivates releasing $\{v_i(\tau)\}$ rather than S, and surfaces a measurement gap that the rest of the paper leverages.

4 The Messier dataset

To construct MESSIER, we combine established public releases with new evaluations targeting benchmarks where per-task data is scarce or absent. The sources include prior capability-measurement releases (METR (Kwa et al., 2026), BRIDGE (Liu et al., 2026), Agent Psychometrics (Ge et al., 2026)), General AgentBench (Li et al., 2026b), public benchmark dumps released by individual benchmark authors, and uniform five-agent reruns we collected through the Harbor harness. Table 1 summarizes the resulting corpus alongside related collections. The source releases are described next.

METR (Kwa et al., 2026) measures agent capacity by task-horizon length, defined as the duration of tasks a model can reliably complete when calibrated against human completion times. They released per-task outcomes for three time-horizon benchmarks (HCAST, SWAA, RE-Bench (Wijk et al., 2024)) used in this measurement, which we ingest directly.

BRIDGE (Liu et al., 2026) extends METR's task-horizon framing with explicit 2PL item-response theory, fitting per-agent ability and per-task difficulty jointly across METR-band tasks. The BRIDGE release contributes new evaluations on MLE-Bench (Chan et al., 2025), GDPval (Patwardhan et al., 2025), and CyBench (Zhang et al., 2025), plus human-time annotations we use for SWE-bench Verified (Jimenez et al., 2024).

Agent Psychometrics (Ge et al., 2026) is a recent effort to formalize psychometric measurement for language-model agents, fitting IRT-style ability and difficulty parameters across a focused set of agent benchmarks. We use their release for per-task outcomes on TerminalBench (Merrill et al., 2026), GSO (Shetty et al., 2026), SWE-bench (Deng et al., 2025), and SWE-bench Pro (Deng et al., 2025).

General AgentBench (Li et al., 2026b) releases agent trajectories spanning search, coding, mathematical-reasoning, and tool-orchestration benchmarks. We use this release for per-task outcomes on MathHay (Wang et al., 2024), BrowseComp (Wei et al., 2025), WebVoyager (He et al., 2024), and MCPBench (Wang et al., 2025b).

Contributed runs. Public data is unevenly available, as trial campaigns are expensive at frontier-model scale, and several recent benchmarks release only aggregate leaderboard scores. To fill these gaps, we contribute a set of six new evaluations on HarveyAI-Lab (Grupen et al., 2026), MedAgentBench (Jiang et al., 2025), DABStep (Egg et al., 2025), QCircuitBench (Yang et al., 2026), ScienceAgentBench (Chen et al., 2025b), and ReplicationBench (Ye et al., 2025). The new evaluations apply a uniform five-agent grid of frontier models under the OpenHands (Wang et al., 2025a) scaffold through the Harbor harness, at a total cost of approximately $10,000. All new runs include full execution traces.

Public benchmark releases. For the majority of benchmarks (HLE (Phan et al., 2025), LiveCodeBench, GDPval, MathArena (Balunovic et al., 2026), SWE-bench, BFCL, OSWorld, τ^2-bench, and others), we obtain per-task outcomes from author-released artifacts, either as Hugging Face datasets or directly from benchmark GitHub repositories. When a primary release omits per-task detail for some agent-task pairs, we cross-reference auxiliary sources, including alternate Hugging Face mirrors and leaderboard CSVs, under normalized identifiers. Appendix A.2 lists full source URLs per benchmark.

After consolidation, the corpus contains 30 benchmarks, 11,891 tasks, 74,205 verifiers, and 957,253 trial outcomes across 345 models and 205 scaffolds (714 distinct agents). The data are summarized in Table 2.

Reconciliation pipeline. The pipeline consists of four stages. First, we cast each incoming record into a uniform data model derived from §3: every environment groups one or more tasks, every task

Benchmark	Benchmark group	Verifier type	Aggregation	Tasks	Verifiers	Outcomes	Agents
SWE-bench	Programming	script	all-pass	1,500	1,500	54 k	92
SWE-bench Verified	Programming	script	all-pass	500	500	129 k	192
SWE-bench Pro	Programming	script	all-pass	731	731	12 k	14
GSO	Programming	script	all-pass	102	102	1.5 k	15
LiveCodeBench	Programming	script	all-pass	1,055	1,055	30 k	22
QCircuitBench	Programming	script	threshold	28	28	0.1 k	5
TerminalBench	Programming	script	all-pass	89	89	62 k	146
CyBench	Programming	exact match	single-verifier	15	15	0.1 k	8
HCAST	Research & reasoning	script	threshold	157	157	15 k	20
SWAA	Research & reasoning	script	single-verifier	66	66	8.3 k	20
RE-Bench	Research & reasoning	script	threshold	5	5	0.5 k	20
MLE-bench	Research & reasoning	script	threshold	82	164	1.5 k	13
ScienceAgentBench	Research & reasoning	script	single-verifier	102	102	0.5 k	5
ReplicationBench	Research & reasoning	script	single-verifier	90	90	0.4 k	5
HLE	Research & reasoning	judge	single-verifier	1,369	1,369	12 k	18
MathArena	Research & reasoning	judge	single-verifier	352	352	51 k	84
MathHay	Research & reasoning	exact match	single-verifier	75	75	1.3 k	5
TheAgentCompany	Enterprise workflows	script	all-pass	175	483	11 k	18
HarveyAI-Lab	Enterprise workflows	judge	all-pass	1,000	60,187	221 k	5
GDPval	Enterprise workflows	judge	threshold	220	220	3.5 k	16
MedAgentBench	Enterprise workflows	script	single-verifier	300	300	1.5 k	5
τ^2-bench	Enterprise workflows	exact match	all-pass	375	2,852	60 k	7
MCPBench	Enterprise workflows	judge	threshold	52	312	6.3 k	5
DABStep	Enterprise workflows	exact match	single-verifier	450	450	2.2 k	5
OSWorld	GUI	script	single-verifier	361	361	31 k	48
OnlineMind2Web	GUI	human	single-verifier	300	300	3.0 k	10
BrowseComp	GUI	judge	single-verifier	124	124	1.8 k	5
WebVoyager	GUI	judge	single-verifier	65	65	1.3 k	5
BFCL-Live	Function calling	exact match	single-verifier	1,351	1,351	147 k	108
BFCL-Multi-Turn	Function calling	exact match	all-pass	800	800	87 k	108
Total	5 groups	4 types	3 rules	**11,891**	**74,205**	**960 k**	**714**

Table 2: **MESSIER per-benchmark summary.** Rows grouped by benchmark group (§4); verifier and aggregation taxonomies defined in §3. The *Verifiers* column counts per-criterion rows for multi-verifier tasks and the task itself otherwise. *Outcomes* counts per-trial scoring events, with one entry per criterion for multi-verifier tasks. HarveyAI-Lab, MedAgentBench, DABStep, QCircuitBench, ReplicationBench, and ScienceAgentBench were collected via Harbor under a uniform five-agent OpenHands grid. Total agents is the union across benchmarks.

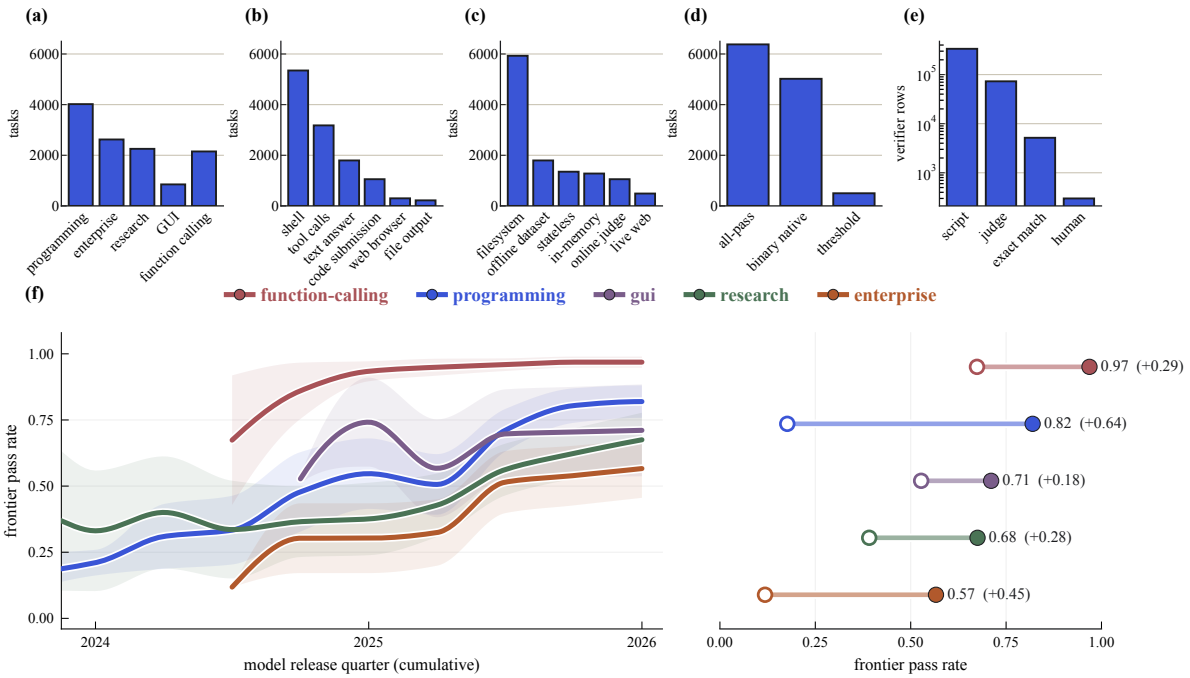

Figure 1: **Composition and frontier analysis.** Panels summarize the corpus across the dimensions defined in §3: (a) tasks by benchmark group, (b) action space, (c) environment state, (d) aggregation rule, and (e) verifier type. Counts in (b)–(e) are on a log scale. Panel (f) tracks frontier pass rate by release quarter and benchmark group, with dumbbell markers connecting each group's first and final observed quarter.

carries one or more typed verifiers (python script, exact-match check, LLM judge, or human label) with an aggregation rule, and each task additionally carries short free-text descriptions of itself, its action space, and its environment state. Second, following Epoch ECI's naming conventions (Ho et al., 2025), we map model identifier, scaffold, and release date into standardized categories. Third, we verify the mapped corpus along two lines: deterministic tests covering identifier hygiene, date validity (e.g. agent date cannot be $<$ benchmark date), exact-match deduplication on `(benchmark, task_id)`, and text quality (HTML entities, control characters). We follow automatic checks with manual spot checks, focusing on newly contributed runs and entries stitched together from multiple sources. Finally, tasks receive SOC occupational (U.S. Bureau of Labor Statistics, 2018) and NAICS (Office of Management and Budget, 2022) industry codes following GDPval (Patwardhan et al., 2025) through a three-voter LLM ensemble with 88.3% (SOC) and 86.8% (NAICS) majority agreement, with the remainder adjudicated by a stronger model (Appendix A.3).

After reconciliation, each score links to its model and scaffold, the environment and task on which it is measured, and the verifiers that determine success. The pipeline itself is organized as one self-contained builder per source, and the 30 existing builders serve as worked examples. Thus, extending the corpus with a new benchmark is a localized contribution that does not touch the core consolidation logic, which keeps the barrier low for both researchers and agent-driven contributions. We organize the 30 benchmarks into five groups by the kind of work the agent performs: programming (software-engineering tasks), research and reasoning (scientific and mathematical problems), enterprise workflows (professional office work), GUI (web and desktop navigation), and function calling (tool selection and invocation). Appendix A.1 describes each group in detail.

5 Analysis & applications

We use MESSIER to study what becomes measurable once agent evaluations are harmonized below the task-level pass rate. The analysis begins with a descriptive account of frontier progress across benchmark groups, then shows that verifier aggregation can change measured capability. We demonstrate two applications of the corpus: reproducing capability indices from open data and predicting task difficulty before running new evaluations.

5.1 Inconsistent frontier progress

Figure 1f provides a descriptive view of frontier progress across the five benchmark groups represented in MESSIER. In quarter q, an agent is eligible if its release date is earlier than q. For each task in a benchmark, we take the highest score among eligible agents. The benchmark frontier is the average of these per-task maxima, and the group frontier averages across the group's benchmarks. This analysis is not intended to estimate field-wide algorithmic progress, since benchmarks, scaffolds, and evaluation sources differ across groups. Instead, the frontier curves identify where aggregate task scores suggest progress and where they may hide finer-grained execution improvements.

The observed frontier rises across all tracked domains between 2024 and 2026, but progress is asymmetric. Function-calling benchmarks approach saturation at a contemporary frontier score of 0.97. Programming follows at 0.82 and shows the largest net growth, with an absolute improvement of 0.64 over the two-year window. Enterprise workflows remain the lowest-performing domain at 0.57. This improves over the historical enterprise baseline, but the low absolute score indicates that multi-step professional workflows remain difficult for agents. Research and reasoning (0.68) and GUI benchmarks (0.71) occupy the intermediate performance band.

Yet a low task-level pass rate does not, by itself, tell us what agents are failing to do. In multi-verifier benchmarks, an all-pass rule assigns an identical zero to a trajectory that satisfies no requirements and to one that satisfies nearly all of them. Thus, aggregate frontier curves may understate partial capability gains precisely in the domains where tasks contain many interacting requirements. This motivates a finer-grained question: when scores are low, do they reflect complete failure, or do they obscure partial progress through aggregation?

5.2 Aggregation obscures partial progress

The verifier-level records in MESSIER let us test whether task-level failures constitute complete failure or partial completion hidden by aggregation. We focus on the three benchmarks for which per-criterion outcomes are available and abundant. In these benchmarks, the official score counts a trial

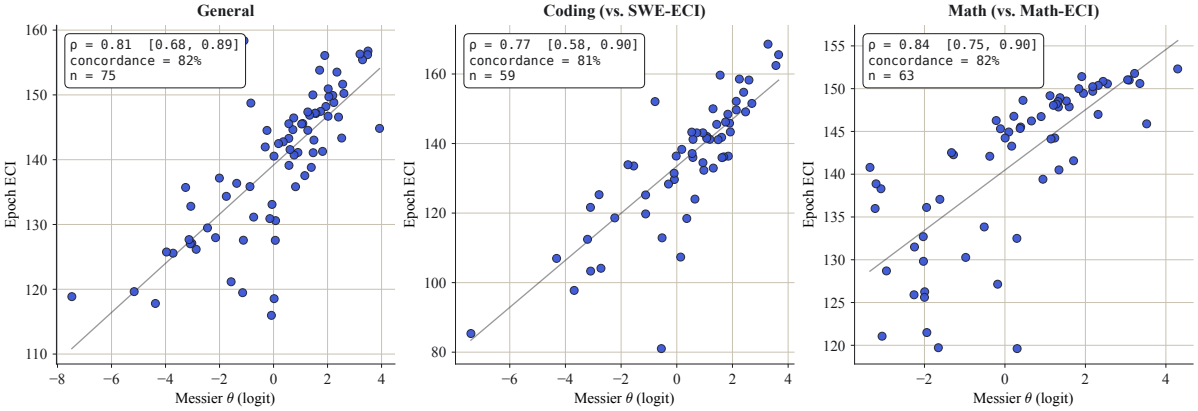

Figure 2: **Capability orderings from MESSIER vs. Epoch's ECI.** Each panel plots MESSIER-fit per-agent θ (logit scale) against Epoch's published ECI for the matched subset, with Spearman ρ, 95% bootstrap CI over 1000 resamples, pairwise concordance, and matched-agent count. The General panel uses Epoch's overall ECI ($n = 75$); Coding uses SWE-ECI ($n = 59$); Math uses Math-ECI ($n = 63$). A 1PL fit on MESSIER alone aligns with Epoch's orderings without any new evaluation campaign.

Benchmark	All-pass	Soft mean	$\geq 50\%$	Near-pass	ρ (soft, all-pass)
HarveyAI-Lab	0.002	0.288	0.298	0.080	0.667
τ^2-bench	0.479	0.754	0.809	0.113	0.857
TheAgentCompany	0.153	0.340	0.358	0.008	0.981

Table 3: **Counterfactual aggregation on benchmarks with per-criterion outcomes.** We rescore the same trials under three alternative rules: all-pass (the official rule), soft (mean fraction of criteria passed), and $\geq 50\%$ (majority pass). *Near-pass* is the fraction of trials that fail all-pass but pass at least 80% of criteria. The last column reports the Spearman rank correlation across agents between their mean all-pass and mean soft scores: $\rho = 1$ would mean the two rules preserve agent ordering; lower values indicate the rules reorder agents.

as successful only when all criteria are satisfied (all-pass). We rescore the same trials under two counterfactual aggregation rules. The first is a soft score, computed as the mean fraction of criteria passed. The second is a majority-pass score, which marks a rollout as successful when at least half of its criteria pass (marked as $\geq 50\%$). We also report a near-pass rate, measuring how often trials fail all-pass grading despite satisfying at least 80% of criteria.

Table 3 shows that aggregation rules affect measured capability. Across all three benchmarks, soft and majority-pass scores are substantially higher than the official all-pass score, indicating that many task-level failures contain partial completion. The effect is largest on HarveyAI-Lab: all-pass grading yields a task success rate of only 0.2%, while agents satisfy 28.8% of criteria on average. Under majority-pass scoring, the measured success rate rises to 29.8%.

Aggregation also reorders agents. On HarveyAI-Lab, per-agent mean all-pass and mean soft scores correlate at Spearman $\rho = 0.667$, well below the ≥ 0.86 on τ^2-bench and TheAgentCompany. The other two benchmarks reorder less because their agents are better separated in absolute capability. Taken together, the counterfactual scores and rank shifts show that task-level pass rates conflate criterion-level performance with aggregation strictness. As a result, aggregate scores can erase substantial partial progress and, in some settings, change the relative ordering of agents.

5.3 Open-data capability scales

We next assess whether MESSIER data are of sufficient quality to track an established cross-benchmark capability ordering (Epoch ECI (Ho et al., 2025)) without launching a new evaluation campaign. To do so, we fit a standard Rasch (1PL IRT) model (Rasch, 1960) to the consolidated agent-task response matrix. To make our estimates comparable to ECI, we follow their model-level aggregation protocol and select the highest-scoring scaffold configuration for each unique *(model, benchmark)* when constructing the response matrix. Under this model, the probability that agent a successfully completes task t is

$$P(Y_{at} = 1 \mid \theta_a, \beta_t) = \frac{1}{1 + e^{\beta_t - \theta_a}}, \qquad (1)$$

where $Y_{at} \in \{0, 1\}$ represents the binary trial outcome, θ_a is the latent agent ability, and β_t is the task difficulty. Full estimation and optimization details are deferred to Appendix B.2.

Because MESSIER focuses on interactive environments rather than static text or multiple-choice benchmarks, its composition differs from Epoch's original dataset pool. We therefore validate the fitted abilities by comparing agent rank orderings to the published ECI rankings using Spearman rank correlation (ρ), rather than comparing absolute ability values. Because benchmarks often mix multiple skills, we use MESSIER's task-level classifications to align the comparison by domain. Specifically, we restrict the 1PL model to tasks mapped to relevant SOC categories, using SOC 15-1200 for programming and SOC 15-2000 for mathematics.

As shown in Figure 2, the resulting 1PL ability estimates (θ_a) closely track the published ECI rankings. Across 75 matched agents, we find rank correlations of $\rho = 0.81$ for general capability, $\rho = 0.77$ for programming, and $\rho = 0.84$ for mathematics. A 2PL fit yields nearly identical agent orderings on the same data, with Spearman $\rho = 0.98$ relative to the 1PL fit (Appendix B.2). Beyond this correspondence with ECI, the scale and structure of MESSIER support custom capability scales across dataset families, occupations, action spaces, and verifier designs.

5.4 Pre-trial task difficulty prediction

As a final application, we test whether MESSIER metadata contains signal about task difficulty before trials are run. Using the fitted IRT task parameters from Section 5.3, we predict two targets: raw difficulty β and within-benchmark z-scored β, following the methodology of Agent Psychometrics (Ge et al., 2026). We fit a RidgeCV regressor on `bge-base-en` sentence embeddings (Xiao et al., 2024) using random k-fold cross-validation, comparing two inputs: the raw instruction text (*task*) and the concatenation of task, environment, action descriptions, and gold answer (termed *all*).

Table 4 shows that the additional metadata yields a modest but consistent improvement across both targets. For raw β, the model only slightly outperforms the benchmark-mean baseline, suggesting that benchmark identity explains much of the signal. For z-scored β, which removes benchmark-level shifts, the model more clearly outperforms the baseline, indicating that task descriptions contain signal about relative difficulty within benchmarks.

Target	Input	Model	Baseline
β (raw)	task	0.484 ± 0.008	0.465
	all	$\mathbf{0.508 \pm 0.009}$	0.465
β (z-scored)	task	0.263 ± 0.015	-0.081
	all	$\mathbf{0.298 \pm 0.011}$	-0.081

Table 4: **Difficulty prediction across two targets and two input configurations.** Targets: raw IRT difficulty β and within-benchmark z-scored β. Inputs: *task* = raw instruction text; *all* = concatenation of task, environment, and action descriptions plus gold answer. The baseline predicts the training-fold benchmark mean. Entries report Spearman ρ under 5-fold cross-validation, mean $\pm$ std across folds. $N = 10{,}406$ tasks.

6 Conclusion

We introduced MESSIER, a unified agent evaluation corpus consolidating 30 benchmarks, 714 agents, 11,891 tasks, and 74,205 verifiers (§4), including new five-agent runs on six benchmarks. By reconciling agents, tasks, environments, verifiers, and aggregation rules into a shared schema, MESSIER makes heterogeneous agent evaluations comparable at both macro- and micro-levels.

Our analysis shows that this added granularity changes what can be measured. Frontier progress is uneven across benchmark groups, with enterprise workflows remaining substantially less solved than programming and function calling. On benchmarks with per-criterion outcomes, counterfactual rescoring shows that all-pass task scores can erase substantial partial progress and sometimes change agent rankings. The same harmonized records also support open psychometric calibration, aligning with Epoch's ECI ordering at Spearman $\rho = 0.81$, and provide useful signal for within-benchmark difficulty prediction.

Looking ahead, MESSIER's verifier-level records and execution traces can support studies of aggregation, judge design, reward hacking, evaluation awareness, and sandbagging at the granularity where these phenomena occur. Its task classifications and difficulty estimates can also help construct targeted evaluations by occupation, domain, action space, and task hardness. By reducing the cost of reusing and extending prior evaluations, MESSIER aims to make future agent evaluations easier to compare, audit, and build upon.

Limitations

Several limitations remain. First, the corpus reflects the current evaluation landscape, which is concentrated in technical, professional, and knowledge-work domains, conducted primarily in English, and largely built on Western software conventions. As a result, findings may not generalize to other languages, cultural settings, or non-technical forms of work. Expanding coverage would require both new benchmarks in underrepresented domains and open per-task releases that make such evaluations available for consolidation.

Second, our collection relies on upstream benchmarks whose construction and quality control are outside our purview. The field is actively pushing the frontier of long-horizon tasks and improving evaluation reliability, as evidenced by the recent emergence of several "-verified" benchmark variants. However, this validation remains an ongoing community effort. For now, we largely assume the validity of these underlying datasets and rely on the broader community to establish their trustworthiness. As these efforts grow, we could systematically track known flawed tasks across benchmarks, which may eventually allow us to model and anticipate these failures.

Third, predicting task difficulty prior to execution remains a significant challenge. Using embeddings for this purpose assumes that difficulty is partly captured by semantic similarity and vector search, even though embeddings primarily encode semantic context rather than inherent complexity. We made this methodological choice deliberately to align with recent foundational work in this area (e.g., Agent Psychometrics (Ge et al., 2026)). Nevertheless, we believe *a priori* difficulty prediction represents an exciting direction for future research, particularly through approaches that leverage the information-theoretic entropy of models rather than relying on semantic embeddings.

Ethical Considerations

As task horizons lengthen (Kwa et al., 2026), the financial and environmental costs of evaluation campaigns grow substantially (Strubell et al., 2019), widening the asymmetry between well-resourced and under-resourced research groups. While MESSIER is designed to alleviate this burden by minimizing redundant evaluation runs, the underlying compute requirements remain a structural equity concern. To partially bridge this gap, we have invested approximately $10,000 in computational resources to contribute this dataset. By standardizing and distributing these trials publicly, we hope to lower the barrier to entry for researchers who lack the capital to run large-scale agent campaigns from scratch.

Furthermore, agent benchmarks frequently abstract compensated human labor. Reading quantitative scores in isolation risks obscuring the socioeconomic implications for the workers whose tasks are being automated. Evaluation releases must therefore maintain a clear accounting of whose labor is being measured and whom that measurement ultimately serves.

Finally, we note that our consolidation strictly adheres to the original licenses, copyright constraints, and terms of use established by the creators of the upstream benchmarks.

References

Yonas Atinafu and Robin Cohen. 2026. Rewardhackingagents: Benchmarking evaluation integrity for llm ml-engineering agents. *arXiv preprint arXiv:2603.11337*.

Mislav Balunovic, Jasper Dekoninck, Ivo Petrov, Nikola Jovanović, and Martin Vechev. 2026. Matharena: Evaluating llms on uncontaminated math competitions. *Advances in Neural Information Processing Systems*, 38.

Victor Barres, Honghua Dong, Soham Ray, Xujie Si, and Karthik Narasimhan. 2025. τ^2-bench: Evaluating conversational agents in a dual-control environment. *arXiv preprint arXiv:2506.07982*.

Antoine Bigeard, Langston Nashold, Rayan Krishnan, and Shirley Wu. 2025. Finance agent benchmark: Benchmarking llms on real-world financial research tasks. *arXiv preprint arXiv:2508.00828*.

Jonathan Bragg, Mike D'Arcy, Nishant Balepur, Dan Bareket, Bhavana Dalvi, Sergey Feldman, Dany Haddad, Jena D Hwang, Peter Jansen, Varsha Kishore, and 1 others. 2025. Astabench: Rigorous benchmarking of ai agents with a scientific research suite. *arXiv preprint arXiv:2510.21652*.

Florian Brand and Jean-Stanislas Denain. 2025. Why benchmarking is hard. Accessed: 2026-05-25.

Ryan Burnell, Wout Schellaert, John Burden, Tomer D Ullman, Fernando Martinez-Plumed, Joshua B Tenenbaum, Danaja Rutar, Lucy G Cheke, Jascha Sohl-Dickstein, Melanie Mitchell, and 1 others. 2023. Rethink reporting of evaluation results in ai. *Science*, 380(6641):136–138.

Jun Shern Chan, Neil Chowdhury, Oliver Jaffe, James Aung, Dane Sherburn, Evan Mays, Giulio Starace, Kevin Liu, Leon Maksin, Tejal Patwardhan, and 1 others. 2025. Mle-bench: Evaluating machine learning agents on machine learning engineering. In *International Conference on Learning Representations*, volume 2025, pages 50466–50494.

Chen Chen, Xinlong Hao, Weiwen Liu, Xu Huang, Xingshan Zeng, Shuai Yu, Dexun Li, Shuai Wang, Weinan Gan, Yuefeng Huang, and 1 others. 2025a. Acebench: Who wins the match point in tool usage? *arXiv preprint arXiv:2501.12851*.

Ziru Chen, Shijie Chen, Yuting Ning, Qianheng Zhang, Boshi Wang, Botao Yu, Yifei Li, Zeyi Liao, Chen Wei, Zitong Lu, and 1 others. 2025b. Scienceagentbench: Toward rigorous assessment of language agents for data-driven scientific discovery. In *International Conference on Learning Representations*, volume 2025, pages 96934–96990.

Xiang Deng, Jeff Da, Edwin Pan, Yannis Yiming He, Charles Ide, Kanak Garg, Niklas Lauffer, Andrew Park, Nitin Pasari, Chetan Rane, and 1 others. 2025. Swe-bench pro: Can ai agents solve long-horizon software engineering tasks? *arXiv preprint arXiv:2509.16941*.

Alex Egg, Martin Iglesias Goyanes, Friso Kingma, Andreu Mora, Leandro von Werra, and Thomas Wolf. 2025. Dabstep: Data agent benchmark for multi-step reasoning. *arXiv preprint arXiv:2506.23719*.

Jonathan Gabor, Jayson Lynch, and Jonathan Rosenfeld. 2025. Evilgenie: A reward hacking benchmark. *arXiv preprint arXiv:2511.21654*.

Johannes Gasteiger, Akbir Khan, Sam Bowman, Misha Wagner, Vladimir Mikulik, Ethan Perez, and Fabien Roger. 2025. Automated researchers can subtly sandbag. Anthropic Alignment Blog.

Chris Ge, Daria Kryvosheieva, Daniel Fried, Uzay Girit, and Kaivalya Hariharan. 2026. Agent psychometrics: Task-level performance prediction in agentic coding benchmarks. *arXiv preprint arXiv:2604.00594*.

Niko Grupen, Gabe Pereyra, and Julio Pereyra. 2026. Open-sourcing :Harvey:'s long horizon legal agent benchmark. Harvey AI Blog.

Hongliang He, Wenlin Yao, Kaixin Ma, Wenhao Yu, Yong Dai, Hongming Zhang, Zhenzhong Lan, and Dong Yu. 2024. Webvoyager: Building an end-to-end web agent with large multimodal models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 6864–6890.

Anson Ho, Jean-Stanislas Denain, David Atanasov, Samuel Albanie, and Rohin Shah. 2025. A rosetta stone for ai benchmarks. *arXiv preprint arXiv:2512.00193*.

Tim Tian Hua, Andrew Qin, Samuel Marks, and Neel Nanda. 2025. Steering evaluation-aware language models to act like they are deployed. *arXiv preprint arXiv:2510.20487*.

Naman Jain, Alex Gu, Wen-Ding Li, Fanjia Yan, Tianjun Zhang, Sida Wang, Armando Solar-Lezama, Koushik Sen, and Ion Stoica. 2025. Livecodebench: Holistic and contamination free evaluation of large language models for code. In *International Conference on Learning Representations*, volume 2025, pages 58791–58831.

Yixing Jiang, Kameron C Black, Gloria Geng, Danny Park, James Zou, Andrew Y Ng, and Jonathan H Chen. 2025. Medagentbench: a virtual ehr environment to benchmark medical llm agents. *Nejm Ai*, 2(9):AIdbp2500144.

Carlos E Jimenez, John Yang, Alexander Wettig, Shunyu Yao, Kexin Pei, Ofir Press, and Karthik Narasimhan. 2024. Swe-bench: Can language models resolve real-world github issues? In *International Conference on Learning Representations*, volume 2024, pages 54107–54157.

Leslie Pack Kaelbling, Michael L Littman, and Anthony R Cassandra. 1998. Planning and acting in partially observable stochastic domains. *Artificial intelligence*, 101(1-2):99–134.

Sayash Kapoor, Benedikt Stroebl, Peter Kirgis, Nitya Nadgir, Zachary S Siegel, Boyi Wei, Tianci Xue, Ziru Chen, Felix Chen, Saiteja Utpala, and 1 others. 2025. Holistic agent leaderboard: The missing infrastructure for ai agent evaluation. *arXiv preprint arXiv:2510.11977*.

Thomas Kwa, Ben West, Joel Becker, Amy Deng, Katharyn Garcia, Max Hasin, Sami Jawhar, Megan Kinniment, Nate Rush, Sydney Von Arx, and 1 others. 2026. Measuring ai ability to complete long software tasks. *Advances in Neural Information Processing Systems*, 38:92213–92266.

Xiangyi Li, Wenbo Chen, Yimin Liu, Shenghan Zheng, Xiaokun Chen, Yifeng He, Yubo Li, Bingran You, Haotian Shen, Jiankai Sun, and 1 others. 2026a. Skillsbench: Benchmarking how well agent skills work across diverse tasks. *arXiv preprint arXiv:2602.12670*.

Xiaochuan Li, Ryan Ming, Pranav Setlur, Abhijay Paladugu, Andy Tang, Hao Kang, Shuai Shao, Rong Jin, and Chenyan Xiong. 2026b. Benchmark test-time scaling of general llm agents. *arXiv preprint arXiv:2602.18998*.

Percy Liang, Rishi Bommasani, Tony Lee, Dimitris Tsipras, Dilara Soylu, Michihiro Yasunaga, Yian Zhang, Deepak Narayanan, Yuhuai Wu, Ananya Kumar, and 1 others. 2022. Holistic evaluation of language models. *arXiv preprint arXiv:2211.09110*.

Fengyuan Liu, Jay Gala, Dzmitry Bahdanau, Siva Reddy, Hugo Larochelle, and 1 others. 2026. Bridge:

Predicting human task completion time from model performance. *arXiv preprint arXiv:2602.07267*.

Xiao Liu, Hao Yu, Hanchen Zhang, Yifan Xu, Xuanyu Lei, Hanyu Lai, Yu Gu, Hangliang Ding, Kaiwen Men, Kejuan Yang, and 1 others. 2024. Agentbench: Evaluating llms as agents. In *International Conference on Learning Representations*, volume 2024, pages 52989–53046.

Nicholas Lourie, Kyunghyun Cho, and He He. 2024. Show your work with confidence: Confidence bands for tuning curves. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 3455–3472.

Xing Han Lù, Amirhossein Kazemnejad, Nicholas Meade, Arkil Patel, Dongchan Shin, Alejandra Zambrano, Karolina Stańczak, Peter Shaw, Christopher J Pal, and Siva Reddy. 2025. Agentrewardbench: Evaluating automatic evaluations of web agent trajectories. *arXiv preprint arXiv:2504.08942*.

Chang Ma, Junlei Zhang, Zhihao Zhu, Cheng Yang, Yujiu Yang, Yaohui Jin, Zhenzhong Lan, Lingpeng Kong, and Junxian He. 2024. Agentboard: An analytical evaluation board of multi-turn llm agents. *Advances in neural information processing systems*, 37:74325–74362.

Lovish Madaan, Aaditya K Singh, Rylan Schaeffer, Andrew Poulton, Sanmi Koyejo, Pontus Stenetorp, Sharan Narang, and Dieuwke Hupkes. 2024. Quantifying variance in evaluation benchmarks. *arXiv preprint arXiv:2406.10229*.

Mike A Merrill, Alexander G Shaw, Nicholas Carlini, Boxuan Li, Harsh Raj, Ivan Bercovich, Lin Shi, Jeong Yeon Shin, Thomas Walshe, E Kelly Buchanan, and 1 others. 2026. Terminal-bench: Benchmarking agents on hard, realistic tasks in command line interfaces. *arXiv preprint arXiv:2601.11868*.

Grégoire Mialon, Clémentine Fourrier, Thomas Wolf, Yann LeCun, and Thomas Scialom. 2024. Gaia: a benchmark for general ai assistants. In *International Conference on Learning Representations*, volume 2024, pages 9025–9049.

Joe Needham, Giles Edkins, Govind Pimpale, Henning Bartsch, and Marius Hobbhahn. 2025. Large language models often know when they are being evaluated. *arXiv preprint arXiv:2505.23836*.

Office of Management and Budget. 2022. *North American Industry Classification System: United States, 2022*. Executive Office of the President, Office of Management and Budget.

Shishir G Patil, Huanzhi Mao, Fanjia Yan, Charlie Cheng-Jie Ji, Vishnu Suresh, Ion Stoica, and Joseph E Gonzalez. 2025. The berkeley function calling leaderboard (bfcl): From tool use to agentic evaluation of large language models. In *Forty-second International Conference on Machine Learning*.

Tejal Patwardhan, Rachel Dias, Elizabeth Proehl, Grace Kim, Michele Wang, Olivia Watkins, Simón Posada Fishman, Marwan Aljubeh, Phoebe Thacker, Laurance Fauconnet, and 1 others. 2025. Gdpval: Evaluating ai model performance on real-world economically valuable tasks. *arXiv preprint arXiv:2510.04374*.

Long Phan, Alice Gatti, Ziwen Han, Nathaniel Li, Josephina Hu, Hugh Zhang, Chen Bo Calvin Zhang, Mohamed Shaaban, John Ling, Sean Shi, and 1 others. 2025. Humanity's last exam. *arXiv preprint arXiv:2501.14249*.

Georg Rasch. 1960. *Probabilistic Models for Some Intelligence and Attainment Tests*. Danmarks Pædagogiske Institut, Copenhagen.

Rylan Schaeffer, Brando Miranda, and Sanmi Koyejo. 2023. Are emergent abilities of large language models a mirage? *Advances in neural information processing systems*, 36:55565–55581.

Tommy Sha and Stella Zhao. 2026. Benchscope: How many independent signals does your benchmark provide? *arXiv preprint arXiv:2603.29357*.

Manish Shetty, Naman Jain, Jinjian Liu, Vijay Kethanaboyina, Koushik Sen, and Ion Stoica. 2026. Gso: Challenging software optimization tasks for evaluating swe-agents. *Advances in Neural Information Processing Systems*, 38.

Aarohi Srivastava, Abhinav Rastogi, Abhishek Rao, Abu Awal Md Shoeb, Abubakar Abid, Adam Fisch, Adam R Brown, Adam Santoro, Aditya Gupta, Adrià Garriga-Alonso, and 1 others. 2023. Beyond the imitation game: Quantifying and extrapolating the capabilities of language models. *Transactions on machine learning research*.

Emma Strubell, Ananya Ganesh, and Andrew McCallum. 2019. Energy and policy considerations for deep learning in nlp. In *Proceedings of the 57th annual meeting of the association for computational linguistics*, pages 3645–3650.

Reuben Tan, Baolin Peng, Zhengyuan Yang, Hao Cheng, Oier Mees, Theodore Zhao, Andrea Tupini, Isar Meijier, Qianhui Wu, Yuncong Yang, and 1 others. 2025. Multimodal reinforcement learning with agentic verifier for ai agents. *arXiv preprint arXiv:2512.03438*.

Kunvar Thaman. 2026. Reward hacking benchmark: Measuring exploits in llm agents with tool use. *arXiv preprint arXiv:2605.02964*.

U.S. Bureau of Labor Statistics. 2018. *2018 Standard Occupational Classification Manual*. U.S. Bureau of Labor Statistics.

Teun Van Der Weij, Felix Hofstätter, Oliver Jaffe, Samuel Brown, and Francis Ward. 2025. Ai sandbagging: Language models can strategically underperform on evaluations. In *International Conference on Learning Representations*, volume 2025, pages 73152–73189.

Lei Wang, Shan Dong, Yuhui Xu, Hanze Dong, Yalu Wang, Amrita Saha, Ee-Peng Lim, Caiming Xiong, and Doyen Sahoo. 2024. Mathhay: An automated benchmark for long-context mathematical reasoning in llms. *arXiv preprint arXiv:2410.04698*.

Xingyao Wang, Boxuan Li, Yufan Song, Frank F. Xu, Xiangru Tang, Mingchen Zhuge, Jiayi Pan, Yueqi Song, Bowen Li, Jaskirat Singh, Hoang H. Tran, Fuqiang Li, Ren Ma, Mingzhang Zheng, Bill Qian, Yanjun Shao, Niklas Muennighoff, Yizhe Zhang, Binyuan Hui, and 5 others. 2025a. OpenHands: An open platform for AI software developers as generalist agents. In *International Conference on Learning Representations (ICLR)*.

Zhenting Wang, Qi Chang, Hemani Patel, Shashank Biju, Cheng-En Wu, Quan Liu, Aolin Ding, Alireza Rezazadeh, Ankit Shah, Yujia Bao, and 1 others. 2025b. Mcp-bench: Benchmarking tool-using llm agents with complex real-world tasks via mcp servers. *arXiv preprint arXiv:2508.20453*.

Jason Wei, Zhiqing Sun, Spencer Papay, Scott McKinney, Jeffrey Han, Isa Fulford, Hyung Won Chung, Alex Tachard Passos, William Fedus, and Amelia Glaese. 2025. Browsecomp: A simple yet challenging benchmark for browsing agents. *arXiv preprint arXiv:2504.12516*.

Hjalmar Wijk, Tao Lin, Joel Becker, Sami Jawhar, Neev Parikh, Thomas Broadley, Lawrence Chan, Michael Chen, Josh Clymer, Jai Dhyani, and 1 others. 2024. Re-bench: Evaluating frontier ai r&d capabilities of language model agents against human experts. *arXiv preprint arXiv:2411.15114*.

Shitao Xiao, Zheng Liu, Peitian Zhang, Niklas Muennighoff, Defu Lian, and Jian-Yun Nie. 2024. C-pack: Packed resources for general chinese embeddings. In *Proceedings of the 47th international ACM SIGIR conference on research and development in information retrieval*, pages 641–649.

Tianbao Xie, Danyang Zhang, Jixuan Chen, Xiaochuan Li, Siheng Zhao, Ruisheng Cao, Toh J Hua, Zhoujun Cheng, Dongchan Shin, Fangyu Lei, and 1 others. 2024. Osworld: Benchmarking multimodal agents for open-ended tasks in real computer environments. *Advances in Neural Information Processing Systems*, 37:52040–52094.

Frank Fangzheng Xu, Yufan Song, Boxuan Li, Yuxuan Tang, Kritanjali Jain, Mengxue Bao, Zora Wang, Xuhui Zhou, Zhitong Guo, Murong Cao, and 1 others. 2026. Theagentcompany: benchmarking llm agents on consequential real world tasks. *Advances in Neural Information Processing Systems*, 38.

Zhangchen Xu, Adriana Meza Soria, Shawn Tan, Anurag Roy, Ashish Sunil Agrawal, Radha Poovendran, and Rameswar Panda. 2025. Toucan: Synthesizing 1.5 m tool-agentic data from real-world mcp environments. *arXiv preprint arXiv:2510.01179*.

Tianci Xue, Weijian Qi, Tianneng Shi, Chan Hee Song, Boyu Gou, Dawn Song, Huan Sun, and Yu Su. 2025. An illusion of progress? assessing the current state of web agents. *arXiv preprint arXiv:2504.01382*.

Eddie Yang and Dashun Wang. 2026. Benchmark illusion: Disagreement among llms and its scientific consequences. *arXiv preprint arXiv:2602.11898*.

Rui Yang, Ziruo Wang, Yuntian Gu, Yitao Liang, and Tongyang Li. 2026. Qcircuitbench: A large-scale dataset for benchmarking quantum algorithm design. *Advances in Neural Information Processing Systems*, 38.

Christine Ye, Sihan Yuan, Suchetha Cooray, Steven Dillmann, Ian LV Roque, Dalya Baron, Philipp Frank, Sergio Martin-Alvarez, Nolan Koblischke, Frank J Qu, and 1 others. 2025. Replicationbench: Can ai agents replicate astrophysics research papers? *arXiv preprint arXiv:2510.24591*.

Andy K Zhang, Neil Perry, Riya Dulepet, Joey Ji, Celeste Menders, Justin Lin, Eliot Jones, Gashon Hussein, Samantha Liu, Donovan Jasper, and 1 others. 2025. Cybench: A framework for evaluating cybersecurity capabilities and risks of language models. In *International Conference on Learning Representations*, volume 2025, pages 25094–25243.

Chenxi Zhang, Ziliang Gan, Liyun Zhu, Youwei Pang, Qing Zhang, and Rongjunchen Zhang. 2026. Finmtm: A multi-turn multimodal benchmark for financial reasoning and agent evaluation. *arXiv preprint arXiv:2602.03130*.

Jitian Zhao, Changho Shin, Tzu-Heng Huang, Satya Sai Srinath Namburi GNVV, and Frederic Sala. 2026. Care: Confounder-aware aggregation for reliable llm evaluation. *arXiv preprint arXiv:2603.00039*.

A Supplementary material

A.1 Benchmark groups

Programming. Programming benchmarks place the agent in a software-engineering environment, typically a source repository or terminal session, and evaluate whether the resulting code or command output satisfies executable checks. The group includes SWE-bench, SWE-bench Verified, and SWE-bench Pro, along with LiveCodeBench and GSO for code generation against test suites, TerminalBench and CyBench for shell-level engineering and capture-the-flag tasks, and QCircuitBench for quantum algorithm design. Although SWE-bench Verified is a curated subset upstream, we treat it as disjoint from SWE-bench to avoid double-counting. Within this group, grading is mostly deterministic: script execution with all-pass aggregation over unit tests for most benchmarks, threshold-based grading for QCircuitBench, and exact-match flag comparison for CyBench. Of these benchmarks, QCircuitBench was manually collected by us, while the remaining benchmarks are consolidated from public releases.

Research and reasoning. Research and reasoning benchmarks ask the agent to solve a scientific, mathematical, or research-engineering problem and to produce either a correct answer or a successful experimental artifact. The group splits between research benchmarks where the outcome is an artifact or analytical solution (HCAST, SWAA, RE-Bench, MLE-Bench, ScienceAgentBench, ReplicationBench), and reasoning benchmarks where the outcome is an answer to a difficult problem in mathematics or general knowledge (HLE, MathArena, MathHay). Of these, ScienceAgentBench and ReplicationBench were collected by us; MLE-Bench is consolidated from the BRIDGE release. Three of the research benchmarks (HCAST, SWAA, RE-Bench) additionally carry per-task human completion-time bands, and verifiers across the group span deterministic scoring functions and LLM judges.

Enterprise workflows. Enterprise workflow benchmarks place the agent in delegated professional or office work, including legal review, clinical workflows, customer support, financial analysis, and general knowledge-work tasks. The group spans workflow simulations with rubric-judged outputs (HarveyAI-Lab, GDPval, MCPBench) and deterministic checks (TheAgentCompany, MedAgentBench, DABStep, τ^2-bench). Of these benchmarks, HarveyAI-Lab, MedAgentBench, and DABStep are public benchmarks for which we collected new runs; the remaining four (GDPval, τ^2-bench, TheAgentCompany, MCPBench) are consolidated from public releases. Several benchmarks in this group decompose tasks into multiple criteria, making it possible to study partial completion even when the final task outcome is binary.

GUI navigation. GUI benchmarks evaluate the ability of the agent to operate in graphical or web environments through clicks, keystrokes, and page navigation, with success measured by the final state of the environment. The group includes OSWorld for desktop applications, OnlineMind2Web for live-web navigation, BrowseComp for web research, and WebVoyager for browsing tasks. All four are consolidated from public releases. Verifier methods split across the group: OSWorld relies on deterministic environment-state checks, BrowseComp and WebVoyager use LLM judges on the final response, and OnlineMind2Web is graded by humans.

Function calling. Function-calling benchmarks evaluate whether agents can select and invoke the appropriate tools, often from large tool sets with overlapping function names, required inputs, and contextual constraints such as user roles or permissions. The group consists of BFCL-Live, which evaluates single-turn function calls by structural AST match against a canonical invocation, and BFCL-Multi-Turn, which evaluates multi-turn conversations over stateful in-memory class registries by matching the agent's per-turn method-call trace against a reference trace turn by turn. Within our taxonomy, grading for both benchmarks is treated as exact match, and both configurations are consolidated from public BFCL releases.

A.2 Sources

Table 5 lists the canonical upstream source for each of the 30 MESSIER benchmarks. *HF* denotes a Hugging Face dataset path; *GitHub* denotes a public repository (pinned commits and dataset revisions are recorded in the companion code release). For benchmarks consolidated from secondary releases (BRIDGE, Agent Psychometrics, General AgentBench), we record the consolidator alongside the upstream owner. Harbor-collected entries are those we evaluated ourselves under the OpenHands scaffold (§4).

Benchmark	Source	Upstream license / access status
SWE-bench	GitHub: princeton-nlp/SWE-bench	MIT
SWE-bench Verified	HF: princeton-nlp/SWE-bench_Verified aux. GitHub: dariakryvosheieva/agent-psychometrics aux. GitHub: McGill-NLP/BRIDGE (human-time)	MIT via SWE-bench repo; HF tag not stated
SWE-bench Pro	HF: ScaleAI/SWE-bench_Pro aux. GitHub: dariakryvosheieva/agent-psychometrics	no explicit standard license identified
GSO	HF: gso-bench/gso aux. GitHub: dariakryvosheieva/agent-psychometrics	MIT
LiveCodeBench	HF: livecodebench/code_generation_lite	CC (HF tag; variant unspecified)
QCircuitBench	Harbor (this work)	MIT
TerminalBench	GitHub: laude-institute/terminal-bench via GitHub: dariakryvosheieva/agent-psychometrics	Apache-2.0
CyBench	GitHub: andyzorigin/cyber-bench via GitHub: McGill-NLP/BRIDGE (results + human-time)	Apache-2.0
HCAST	GitHub: METR/hcast-public runs via GitHub: METR/eval-analysis-public	MIT[†]
SWAA	runs via GitHub: METR/eval-analysis-public	no explicit standard license identified
RE-Bench	GitHub: METR/RE-Bench runs via GitHub: METR/eval-analysis-public	MIT[†]
MLE-Bench	Kaggle competitions via GitHub: McGill-NLP/BRIDGE	MIT for code; competition data subject to Kaggle terms
ScienceAgentBench	Harbor (this work)	MIT
ReplicationBench	Harbor (this work)	MIT
HLE	HF: cais/hle aux. GitHub: supaihq/hle (judged-pro labels)	MIT; gated; outcomes only[‡]
MathArena	HF: MathArena/{slug}_outputs	CC-BY-NC-SA-4.0
MathHay	HF: cx-cmu/agent_trajectories	gated; no explicit HF license identified[§]
TheAgentCompany	GitHub: TheAgentCompany/TheAgentCompany GitHub: TheAgentCompany/experiments	MIT for benchmark repo; no explicit license identified for experiments repo
HarveyAI-Lab	Harbor (this work)	no explicit standard license identified
GDPval	HF: openai/gdpval aux. GitHub: McGill-NLP/BRIDGE	no explicit standard license identified
MedAgentBench	Harbor (this work)	MIT
τ^2-bench	GitHub: sierra-research/tau2-bench submissions: Sierra S3	MIT
MCPBench	HF: cx-cmu/agent_trajectories	gated; no explicit HF license identified[§]
DABStep	Harbor (this work)	CC-BY-4.0
OSWorld	GitHub: xlang-ai/OSWorld HF: xlangai/ubuntu_osworld_verified_trajs	Apache-2.0 for code; MIT for HF trajectories
OnlineMind2Web	HF: osunlp/Online-Mind2Web labels: osunlp/Online_Mind2Web_Leaderboard (HF Spaces)	CC-BY-4.0; gated
BrowseComp	HF: cx-cmu/agent_trajectories	gated; no explicit HF license identified[§]
WebVoyager	HF: cx-cmu/agent_trajectories	gated; no explicit HF license identified[§]
BFCL-Live	GitHub: ShishirPatil/gorilla (berkeley-function-call-leaderboard)	Apache-2.0
BFCL-Multi-Turn	GitHub: ShishirPatil/gorilla (berkeley-function-call-leaderboard)	Apache-2.0

Table 5: Per-benchmark sources and upstream license/access status. Group ordering follows Table 2. *aux.* marks secondary releases used to cross-reference or fill per-task gaps under normalized identifiers. The license/status column reports the terms identified for the cited upstream source; it does not define the license of MESSIER as a whole. Our original MESSIER code, schema, mappings, and derived records are released under MIT, without overriding upstream licenses or access conditions. [†]MIT with author requests to avoid publishing unprotected solutions or including tasks in LLM training data. [‡]HLE is tagged MIT on Hugging Face but is gated and asks users not to publicly share, re-upload, or distribute the dataset; we therefore redistribute only derived pass/fail outcomes. [§]The cited Hugging Face trajectory release is gated and no explicit HF license tag was identified; underlying benchmark projects may have separate licenses, but we do not treat those as relicensing the trajectory release.

A.3 SOC and NAICS classification

Tasks receive SOC occupational and NAICS industry codes through a three-tier pipeline. First, three voter models (Anthropic Claude Haiku, Anthropic Claude Sonnet, and OpenAI GPT-5-mini) independently classify each task against the full BLS 2018 SOC structure and Census 2022 NAICS sector list, prompted with the task description and a brief gloss of every code. Second, splits among the three voters are adjudicated by Anthropic Claude Opus on the disputed subset. Third, content-keyword overrides catch a small residue of edge cases. Benchmarks whose tasks deterministically map to a single occupation (e.g., the SWE-bench family to Computer Occupations) bypass the voter pipeline and are crosswalked directly. Across the 4,311 items rated by all three voters, a 2-of-3 majority exists on 88.3% of SOC and 86.8% of NAICS labels; the remaining 11.7% and 13.2% are resolved by Opus as adjudicator. The classifier output schema, voter prompts, and per-stage caches are released alongside the corpus.

A.4 Data model

Figure 3 summarizes the entity-relationship structure of MESSIER, following the framework introduced in §3.

A.5 Contributed-run configuration

The six contributed benchmarks (HarveyAI-Lab, MedAgentBench, DABStep, QCircuitBench, ScienceAgentBench, and ReplicationBench) were collected under a single fixed configuration through the Harbor harness against hosted model APIs; no local GPU training was performed. The agent grid consists of five frontier models held constant across all six benchmarks: `openai/gpt-5`, `openai/gpt-5-nano`, `openai/gpt-4o`, `anthropic/claude-haiku-4-5`, and `anthropic/claude-sonnet-4-5`. Each (agent, task) pair is run once; we do not collect multiple trials per cell. The scaffold is `openhands-sdk` v1.22.0, executed inside Modal sandbox images. Sampling parameters are left at provider defaults. For HarveyAI-Lab's rubric grading, the verifier is an LLM judge (`anthropic/claude-sonnet-4-6`) that issues an independent binary judgment for each of the 33–65 criteria attached to a task, with parent-task success obtained by all-pass aggregation across criteria. The total API spend was approximately $10,000.

A.6 Licensing

MESSIER is a mixed-license collection assembled from third-party benchmarks with heterogeneous licenses and access conditions. We do not claim that third-party benchmark materials are relicensed under a single umbrella license. Each upstream component remains subject to its original terms, and Table 5 lists the public source and the corresponding license or access status we identified at the time of release.

Our original contributions in this work, including normalization code, schema definitions, benchmark mappings, and released derived records such as summary statistics or outcome tables, are licensed under the MIT License. This license applies only to our original contributions and does not override or replace any upstream license.

When an upstream source is gated, lacks a clear public license, or presents ambiguous redistribution terms, we do not redistribute the underlying task content. Instead, where permitted, we release only minimal derived artifacts such as normalized identifiers and/or pass/fail outcomes, and direct users to obtain the original data from the upstream provider.

B Methodology

B.1 Human-time calibration

Under BRIDGE's exact methodology (METR-band tasks, 2PL IRT), MESSIER's pipeline aligns with their log-linear slope within the bootstrap CI: slope $= 0.80$, 95% CI $[0.71, 0.90]$, $n = 158$. The corresponding doubling factor per unit of latent capability is 2.23, matching BRIDGE's published 2.26 and METR's 2.36. We report this as an implementation check confirming pipeline consistency on overlapping data.

B.2 Item response theory estimation

We fit the 1PL Rasch model $P(Y_{at} = 1) = \sigma(\theta_a - \beta_t)$ via stochastic variational inference in Pyro, with normal priors on θ_a and β_t (mean 0, standard deviation 1) and mean-field normal guides. Optimization uses ClippedAdam with learning rate 0.01 over 1000–2000 epochs (seed 42); for the BRIDGE-aligned replication (Appendix B.1) we additionally fit a 2PL variant, $P(Y_{at} = 1) = \sigma(\alpha_t(\theta_a - \beta_t))$, with a log-normal discrimination prior on α_t; on the full corpus, the 1PL and 2PL agent orderings agree at Spearman $\rho = 0.98$ ($n = 344$ agents), so we report 1PL throughout. Items with constant outcomes across all agents

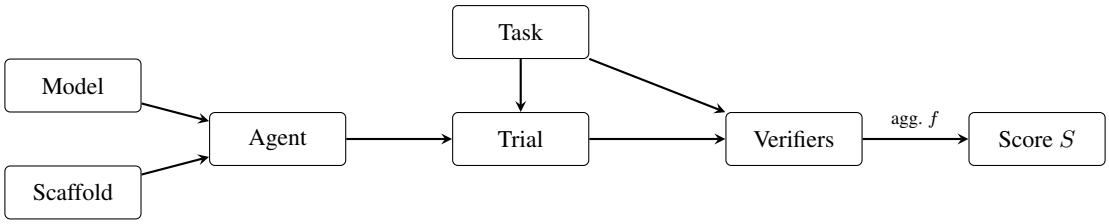

Figure 3: Entity-relationship view of MESSIER. An *agent* is the composition of a *model* and a *scaffold*. A *task* specifies one or more *verifiers* $v_1, \ldots, v_N$. A *trial* is one execution of an agent on a task; each verifier produces an outcome $v_i(\tau) \in \{0, 1\}$, and the aggregation rule f combines these into the *trial score* S.

(uniformly passed or uniformly failed) are dropped before fitting, since they carry no information for ability estimation. Domain-restricted ability estimates (e.g. for SOC 15-1200 coding tasks) are obtained by holding β_t fixed at its full-corpus posterior mean and refitting θ_a on the domain subset only, which preserves cross-domain comparability of the ability scale.

B.3 Fitted IRT parameter distributions

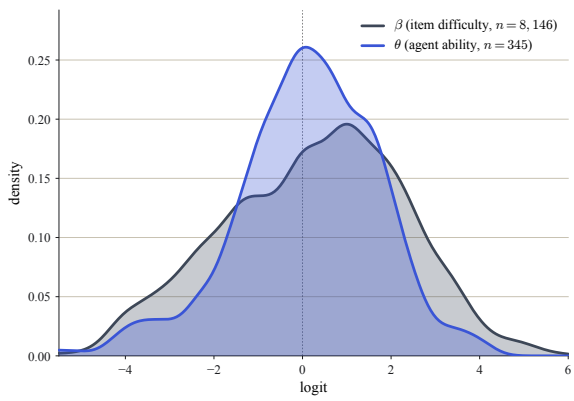

Figure 4: Distributions of fitted IRT 1PL agent ability θ and task difficulty β across the full MESSIER corpus.

B.4 ECI alignment metrics

We measure agreement between MESSIER-fit IRT abilities and Epoch's published ECI through Spearman rank correlation,

$$\rho = 1 - \frac{6 \sum_i d_i^2}{n(n^2 - 1)}, \tag{2}$$

where d_i is the difference in ranks for agent i across the two scales. We report 95% confidence intervals via percentile bootstrap with 1000 resamples of the matched (θ, ECI) pairs.

We also report pairwise concordance: for each pair of distinct matched models (i, j) with $\theta_i \neq \theta_j$ and $\text{ECI}_i \neq \text{ECI}_j$, concordance is the fraction of pairs in which MESSIER and Epoch agree on ordering. Concordance is robust to scale and ties.

For cross-source matching, model identifiers are normalized via the same alias map used by Epoch ECI before computing Spearman or concordance.